AMENDMENTS TO THE CLAIMS

The listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims

1. (currently amended): A ~~bypass system for a data cache~~ multi-streaming microprocessor core, for executing instruction streams running within the multi-streaming microprocessor core at any time, the multi-streaming microprocessor core comprising:

instruction queues, each corresponding to each of the instruction streams, said each of said instruction queues comprising:

first instructions, for dispatch to one or more functional units;

store instructions, for dispatch to a data cache, wherein said store instructions direct write operations; and

load instructions, for dispatch to said data cache, wherein said load instructions direct read operations;

a bypass structure within said data cache, for receiving said store instructions, said bypass structure comprising multiple elements, wherein, if said write operations hit in said data cache, data corresponding to said write operations are stored in one or more of said elements in said bypass structure before said data is written to said data cache; and

address matching logic, coupled to said bypass structure within said data cache, for receiving said load instructions, wherein said read operations use said address matching logic to search said elements of said bypass structure to identify and use any one or more of said elements representing more recent data than that stored in said data cache.

~~two ports to the data cache;~~

~~registers for multiple data entries;~~

~~a bus connection for accepting read and write operations to the cache; and~~

~~address matching and switching logic;~~

~~characterized in that write operations that hit in the data cache are stored as elements in the bypass structure before the data is written to the data cache, and read operations use the address matching logic to search the elements of the bypass structure to identify and use any one or more of the entries representing data more recent than that stored in the data cache memory array, such that a subsequent write operation may free a memory port for a write stored in the bypass structure to be written to the data cache memory array.~~

2. (currently amended): The ~~bypass system of claim 1 wherein~~multi-streaming microprocessor core as recited in claim 1, wherein ~~the memory~~ said read operations and said write operations are limited to 32 bits, and ~~there~~ wherein said elements comprise six elements~~are six distinct entries in the bypass system~~.

3. (currently amended): A multi-streaming microprocessor core, for executing instruction streams running within the multi-streaming microprocessor core at any time, the multi-streaming microprocessor core comprising:

instruction queues, each corresponding to each of the instruction streams, said each of said instruction queues comprising:

first instructions, for dispatch to one or more functional units;

store instructions, for dispatch to a data cache, wherein said store instructions direct write operations; and

load instructions, for dispatch to said data cache, wherein said load instructions direct read operations;

a bypass structure within said data cache, for receiving said store instructions, said bypass structure comprising multiple elements, wherein, if said write operations hit in said data cache, data corresponding to said write operations are stored in one or more of said elements in said bypass structure before said data is written to said data cache;

address matching logic, coupled to said bypass structure within said data cache, for receiving said load instructions, wherein said read operations use said address matching logic to search said elements of said bypass structure to identify and use any one or more of said elements representing more recent data than that stored in said data cache; and

switching logic, coupled to said bypass structure within said data cache, for determining where a newest version of said more recent data resides based on bytes, and wherein one of said read operations matches on multiple elements of said bypass structure.

~~data cache system comprising:~~

~~a data cache memory array; and~~

~~a bypass system connected to the data cache memory array by two ports, and to a bus for accepting read and write operations to the system, and having address matching and switching logic;~~

~~characterized in that write operations that hit in the data cache are stored as elements in the bypass structure before the data is written to the data cache, and read operations use the address matching logic to search the elements of the bypass structure to identify and use any one or more of the entries representing data more recent than that stored in the data cache memory array, such that a subsequent write operation may free a memory port for a write stored in the bypass structure to be written to the data cache memory array.~~

4. (currently amended): The multi-streaming microprocessor core as recited in ~~system of~~ claim 3, wherein ~~the memory~~ said read operations and said write operations are limited to 32 bits, and wherein said elements comprise six elements~~there are six distinct entries in the bypass system~~.

5. (currently amended): A method for eliminating stalls in read and write operations to a data cache within a multi-streaming microprocessor core, comprising~~ steps of~~:

providing multiple instruction streams to corresponding instruction queues, said providing comprising:

first dispatching first instructions to one or more functional units;

second dispatching store instructions to a data cache, wherein the store instructions direct write operations; and

third dispatching load instructions to the data cache, wherein the load instructions direct read operations;

first receiving the store instructions in a bypass structure within the data cache, wherein the bypass structure comprises multiple elements, and wherein, if the write operations hit in the data cache, storing data corresponding to the write operations in one or more of the elements in the bypass structure before the data is written to the data cache; and

second receiving the load instructions in address matching logic within the data cache, wherein the read operations use the address matching logic to search the elements of the bypass structure to identify and use any one or more of the elements representing more recent data than that stored in the data cache.

~~(a) implementing a bypass system having multiple entries and switching and address matching logic, connected to the data cache memory array by two ports and to a bus for accepting read and write operations;~~

~~(b) storing write operations that hit in the cache as entries in the bypass structure before associated data is written to the cache;~~

~~(c) searching the bypass structure entries by read operations, using the address matching and switching logic to determine if entries in the bypass structure represent newer data that that available in the data cache memory array; and~~

~~(d) using the opportunity of a subsequent write operation to free a memory port for simultaneously writing from the bypass structure to the memory array.~~

6. (currently amended): The method ~~of claim 5~~as recited in claim 5, wherein the read and write ~~memory~~ operations are limited to 32 bits, and wherein the multiple elements comprise six elements~~there are six distinct entries in the bypass system~~.

7. (new): The multi-streaming microprocessor core as recited in claim 1, wherein said instruction queues comprise eight instruction queues, each corresponding to each of eight instruction streams.

8. (new): The multi-streaming microprocessor core as recited in claim 7, wherein up to two of said read operations are dispatched to said data cache across all of said eight instruction streams in each cycle.

9. (new): The multi-streaming microprocessor core as recited in claim 7, wherein up to two of said write operations are dispatched to said data cache across all of said eight instruction streams in each cycle.

10. (new): The multi-streaming microprocessor core as recited in claim 1, wherein said data cache is dual ported.

11. (new): The multi-streaming microprocessor core as recited in claim 1, wherein said instruction streams are based on the MIPS instruction set architecture.

12. (new): The multi-streaming microprocessor core as recited in claim 1, wherein one of said read operations matches on more than one of said elements of said bypass structure, and wherein switching logic determines a newest version of a given item of said data based on bytes.

13. (new): The multi-streaming microprocessor core as recited in claim 12, wherein said one of said read operations gets its value from different locations, some of which are in said data cache and some of which are in said bypass structure.

14. (new): The multi-streaming microprocessor core as recited in claim 3, wherein said instruction queues comprise eight instruction queues, each corresponding to each of eight instruction streams.

15. (new): The multi-streaming microprocessor core as recited in claim 14, wherein up to two of said read operations are dispatched to said data cache across all of said eight instruction streams in each cycle.

16. (new): The multi-streaming microprocessor core as recited in claim 14, wherein up to two of said write operations are dispatched to said data cache across all of said eight instruction streams in each cycle.

17. (new): The multi-streaming microprocessor core as recited in claim 3, wherein said instruction streams are based on the MIPS instruction set architecture.

18. (new): The method as recited in claim 5, wherein said providing comprises:

issuing eight instruction streams to eight corresponding instruction queues.

19. (new): The method as recited in claim 18, wherein up to two of the read operations are dispatched to the data cache across all of the eight instruction streams in each cycle.

20. (new): The method as recited in claim 18, wherein up to two of the write operations are dispatched to the data cache across all of the eight instruction streams in each cycle.

21. (new): The method as recited in claim 5, wherein said providing comprises:

basing the instruction streams on the MIPS instruction set architecture.

22. (new): The method as recited in claim 5, wherein one of the read operations matches on more than one of the elements of the bypass structure, and wherein switching logic determines a newest version of a given item of the data based on bytes.

23. (new): The method as recited in claim 22, wherein the one of the read operations gets its value from different locations, some of which are in the data cache and some of which are in the bypass structure.